Data::Downloader

Brian Duggan[1], Curt Tilmes[2], Phil Durbin[1]

[1]ADNET Systems

[2]NASA/GSFC

ESIP Summer Meeting, 2010

brian.duggan@nasa.gov

Workflow

1. Satellite data files enter the the Atmospheric Composition Processing System (ACPS).
2. Algorithms run and produce files.
3. **Data::Downloader** copies files to NFS mounted disks.
4. Scientists improve algorithms using these files.
5. Algorithms are released back into the ACPS.
6. ACPS now produces better files.
7. Go to step 3.

Features

- Replace old files (Unique Resource Names)
- Ensure integrity (MD5 hash)
- Distribute evenly among NFS disks
- Use file metadata from RSS/Atom feeds to organize into directories
- Support arbitrary file metadata
- Lightweight (sqlite)
- Manage repository size with pluggable algorithms (LRU)
- Can be used as a caching layer on minions

Sample situations

- Download 59,308 files.
- In a reasonable amount of time.
- Download 120,128 files, some of which are the same.
- Don't download the same file twice.
- Organize as <year>/<month>/<day>/<filename>.
- The dates have been recalculated, adjust the directories.
- Make one big directory with all the files.
- Newer algorithms are running, replace obsolete files.
- Download files generated since yesterday.
- Download files matching some criteria.
- Overcome NFS issues.
- Identify processing/storage issues.
- Identify missing/corrupt files.

Data::Downloader – Quick Start

Install

```
curl -L http://cpanmin.us > ~/bin/cpanm
chmod +x cpanm
curl -u username -L http://macuv.gsfc.nasa.gov/dist/String-Template/latest.tgz > st.tgz
curl -u username -L http://macuv.gsfc.nasa.gov/dist/Data-Downloader/latest.tgz > dd.tgz
cpanm st.tgz
cpanm dd.tgz
# (also see local::lib)
```

Configure

```
dado config init --filename etc/mirador.yml     # or..
dado config init --filename etc/omi.yml         # or..
dado config init --filename etc/modaps.yml      # or..
dado config init --filename etc/flickr.yml
```

Use

```
dado feeds refresh
dado files download
```

Data::Downloader – Advanced

```
dado config init --filename etc/mirador.yml
dado feeds refresh --dataSet AIRX2RET.005 --starttime '2006-06-01' --endtime '2006-06-02'
dado files --filename 'like: AIR%' dump
dado files --filename 'like: AIR%' dump

dado config update --filename etc/omi.yml
dado feeds --name omi refresh --esdt OMTO3
dado files --esdt OML1BCAL --orbit '[ge: 1000, le: 2000]' dump

dado files --esdt OML1BCAL --md5 'like: a%' download
dado files --esdt OML1BCAL --md5 'like: b%' download

dado --debug root feeds refresh
dado --trace root feeds refresh
dado --progress_bars linktrees rebuild

dado config update --filename etc/flickr.yml
dado feeds --name flickr refresh --tags nasa
dado files download

dado disks --root '/disk12/data' usage
dado repositories dump_stats
```

Configuration : mirador

```
name: mirador
storage_root: /tmp/dado/mirador/store
file_url_template: ''
cache_strategy: Keep

feeds:
  name: mirador
  feed_template: 'http://mirador.gsfc.nasa.gov/cgi-bin/mirador/granlist.pl?dataSet=<dataSet>&
  feed_parameters: [
     { name: dataSet,       default_value: 'AIRX2RET.005'   },
     { name: pointLocation, default_value: '-120,20,-90,50' },
     { name: starttime,     default_value: '2006-06-01T00:00:00Z' },
     { name: endtime,       default_value: '2006-06-11T00:00:00Z' },
     { name: maxgranules,   default_value: 2 },
    ]
  file_source:
    filename_xpath: default:title
    url_xpath: default:id
  metadata_sources:
    - name: start_time
      xpath: time:start
    - name: stop_time
      xpath: time:stop

linktrees:
    - root: /tmp/dado/mirador/data
      path_template: <start_time:%Y/%m/%d>
```

Configuration : modaps

```
name: modaps
storage_root: /tmp/dado/modaps/store
file_url_template: 'ftp://ladsweb.nascom.nasa.gov/allData/<collection>/<product>/<year>/<daynu
cache_strategy: Keep

feeds:
  name: modaps
  feed_template: 'http://modwebsrv.modaps.eosdis.nasa.gov/axis2/services/MODAPSservices
    /getOpenSearch?products=<product>&collection=<collection>&start=<start>&stop=<stop>&
    bbox=<bbox>&coordsOrTiles=<coordsOrTiles>&dayNightBoth=<dayNightBoth>'
  feed_parameters: [
    { name: product,       default_value: 'MOD021KM'   },
    { name: collection,    default_value: 5            },
    { name: start,         default_value: '2009-09-01' },
    { name: stop,          default_value: '2009-09-02' },
    { name: bbox,          default_value: '10,17,35,0' },
    { name: coordsOrTiles, default_value: 'tiles'      },
    { name: dayNightBoth,  default_value: 'DNB'        }]
  file_source:
    filename_xpath: title
    url_xpath: id
  metadata_sources:
    - name: summary
      xpath: summary
metadata_transformations: [
    { input: summary, output: starttime, function_name: extract,
      function_params: 'startTime (\d{4}-\d{2}-\d{2})', order_key: 100 },
    { input: summary, output: product, function_name: extract,
      function_params: '(\S+) format', order_key: 200 }
linktrees:
    - root: /tmp/dado/modaps/data
      path_template: <starttime:%Y/%m/%d>
      condition: ~
```

Configuration : omi

```
name: omi
storage_root: /tmp/dado/omi/store
file_url_template: 'https://omisips1.omisips.eosdis.nasa.gov:8000/data/<md5>/<filename>'
cache_strategy: Keep
feeds:
  name: omi
  feed_template: 'https://omisips1.omisips.eosdis.nasa.gov:8000/service/georss?as=<archiveset
  feed_parameters: [
     { name: count,                      default_value: 10000     },
     { name: startproductiontime,        default_value: ~         },
     { name: startproductiontime_offset, default_value: ~         },
     { name: endproductiontime,          default_value: ~         },
     { name: archiveset,                 default_value: 10003     },
     { name: met,                        default_value: 1         },
    ]
  file_source:
    filename_xpath: datacasting:filename
    md5_xpath: datacasting:md5
    url_xpath: default:link
    urn_xpath: datacasting:unique_identifier
  metadata_sources: [
    { name: archivesets, xpath: datacasting:archivesets },
    { name: starttime,   xpath: datacasting:starttime   },
    { name: esdt,        xpath: datacasting:esdt        },
    { name: orbit,       xpath: datacasting:orbit       } ]
metadata_transformations: [
    { input: archivesets, output: archiveset, function_name: split, order_key: 19 }
linktrees:
  - root: /tmp/dado/omi/data/default
    condition: ~
    path_template: '<archiveset>/<esdt>/<starttime:%Y/%m/%d>'
```

Configuration : flickr

```
name              : flickr
storage_root      : /tmp/flickr_store
file_url_template: 'http://<host>.static.flickr.com/<url1>/<url2>_<size>.<format>'
cache_strategy    : Keep
feeds:
  name            : flickr
  feed_template   : 'http://api.flickr.com/services/feeds/photos_public.gne?tags=<tags>&lang=en
  file_source:
       filename_xpath : 'media:content/@url'
       filename_regex : '/([^/]*)$'
       url_xpath      : 'media:content/@url'
  metadata_sources:
      - name  : date_taken
        xpath : 'dc:date.Taken'
      - name  : 'tags'
        xpath : 'media:category'
metadata_transformations:
    - input: tags
      output: tag
      function_name: split
      order_key: 10
linktrees:
  - root          : /tmp/flickr_by_date
    condition     : ~
    path_template : "<date_taken:%Y/%m/%d>"
  - root          : /tmp/flickr_by_year_tag
    condition     : ~
    path_template : "<date_taken:%Y>/<tag>"
```

Demo

```
dado config init --filename etc/mirador.yml
dado feeds refresh
dado files download
ls /tmp/dado/mirador/data

dado config init --filename etc/flickr.yml
dado --trace root feeds --name flickr refresh --tags nasa
dado files download
tree /tmp/flickr_by_year_tag
```